

An efficient decoding algorithm for stabilizer codes

Adrian Hutter, James R. Wootton, and Daniel Loss

Department of Physics, University of Basel, Klingelbergstrasse 82, CH-4056 Basel, Switzerland

(Dated: February 13, 2013)

To date, the best classical algorithm for performing error correction in the surface code has been minimum-weight perfect matching. However, in this work we present a Markov chain Monte Carlo algorithm that achieves significantly lower logical error rates. It therefore allows any target logical error rate to be obtained using a significantly smaller code. This increase in performance does come at the cost of an increased runtime complexity, but only by a polynomial factor $O(L^\varepsilon)$ for $\varepsilon < 2$. Our algorithm is based on an analytically exact rewriting of the probability of each logical equivalence class, which also suggests that for arbitrary stabilizer codes error correction can be performed to arbitrary accuracy in a runtime $O(\text{poly}(L))$. It is applicable to any stabilizer code, allows for parallelization, and can be used to correct in the case of imperfect stabilizer measurements.

INTRODUCTION

An important primitive for the *processing* of quantum information is the ability to *store* it despite constant corruptive influence of the external environment on the applied hardware and imperfections of the latter. While one approach seeks to achieve this by constructing a *self-correcting quantum memory* (see Ref. [1] for a recent review), an alternative possibility is to dynamically protect the stored quantum information by constantly pumping entropy out of the system. Topological quantum error correction codes [2, 3] store one logical qubit in a large number of physical qubits, in a way which guarantees that a sufficiently low density of errors on the physical qubits can be detected and undone, without affecting the stored logical qubit. Most promising is the surface code [4–7], which requires only local four-qubit parity operators to be measured. While proposals for direct measurement of such operators exist [8–10], most of the literature focuses on time-dependent interactions between the four qubits and an auxiliary qubit, allowing to perform sequential CNOT gates and to finally read the measurement result off the auxiliary qubit. See Ref. [11] for a recent review.

In order to decode the syndrome information, i.e. use the outcomes of all four-qubit measurements to find out how to optimally perform error correction, a classical computation is necessary. This classical computation is not trivial and brute force approaches are infeasible. Decoding algorithms based on renormalization techniques [12] or minimum-weight perfect matching (MWPM) [13] have a runtime complexity $O(L^2)$ and can be parallelized to $O(L^0)$ (neglecting logarithms), where L is the linear size of the code. As these algorithms are approximative, the logical error rates achievable with them fall short of those theoretically achievable by brute force decoding. A Markov chain Monte Carlo (MCMC) algorithm [14] can cope with higher physical error rates than the two mentioned algorithms, but has super-polynomial (yet sub-exponential) runtime complexity. In this work, we present an efficient MCMC decoding algorithm that al-

lows to achieve logical error rates lower than those achievable by means of MWPM [15]. Equivalently, a smaller code size is required to achieve a certain target logical error rate. Our algorithm allows for trade-offs between runtime and achieved logical error rate. We define its runtime to be the minimal computation time such that the achieved logical error rate is lower than the one achievable by means of MWPM. Empirically, we find a runtime complexity $O(L^\alpha)$ with $\alpha < 4$ depending on the error model. The runtime complexity can be parallelized to $O(L^{\alpha-2})$, thus adding complexity $O(L^{\alpha-2})$ to MWPM and renormalization technique decoding algorithms. In summary, in comparison to alternative algorithms [12, 13] our algorithm allows for lower quantum information error rates and smaller code sizes at the cost of a higher classical runtime complexity. Given the current state of the art of quantum and classical information processing, shifting requirements from quantum to classical seems desirable. Our algorithm is generalizable to the (realistic) case of imperfect stabilizer measurements, though we restrict numerical simulations in this work to the case of perfect measurements for simplicity.

ERROR CORRECTION IN SURFACE CODES

Stabilizer operators are, in the context of the surface code, tensor products of σ^x or σ^z operators (see Fig. 1) which are required to yield a +1 eigenvalue when applied to the quantum state stored in the code. Eigenvalues -1 are treated as errors and interpreted as the presence of an *anyon*. A surface code of size L has $n_{\text{stab}} = 2L(L-1)$ (3- and 4-qubit) stabilizers. Since all stabilizers commute, they can be measured simultaneously and hence the presence of anyons can be detected. Any Pauli operator σ^x , σ^y , or σ^z applied to a data qubit creates at least one anyon as it anti-commutes with at least one stabilizer. We call violated σ^x -stabilizers s-anyons and violated σ^z -stabilizers p-anyons.

Given some anyon configuration A , the goal is to apply a series of single-qubit σ^x and σ^z operators, such

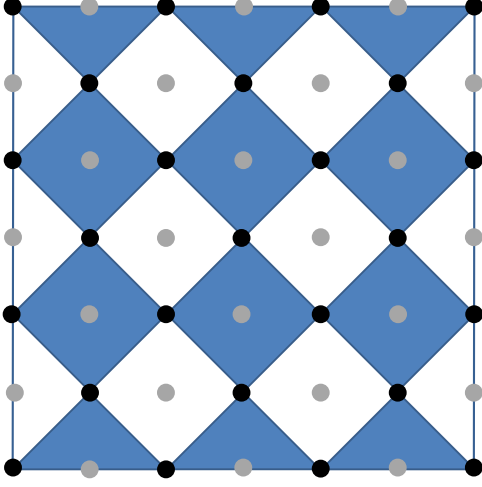


FIG. 1. An $L = 4$ surface code. Black dots are data qubits, grey dots are syndrome qubits that allow to read off the results of the stabilizer measurements when sequential CNOT gates have been performed between them and the adjacent data qubits. Stabilizer operators are either tensor products of σ^x operators (acting on the data qubits around a white square/triangle) or tensor products of σ^z operators (acting on the data qubits around a blue square/triangle).

that all anyons are removed and a trivial operation has been performed on the code subspace. Two such hypotheses about what errors the physical qubits have suffered are equivalent if they can be deformed into each other through the application of stabilizers. Equivalent error chains will lead to the same operation performed on the code subspace (consisting of the states which are $+1$ eigenstates of each stabilizer). For the surface code, there are four such equivalence classes. The goal is therefore to find the most probable equivalence class of error chains and not to find the most likely error chain. The most likely error chain need not be an element of the most likely equivalence class, though trying to correct by undoing the most likely error path is a reasonable approximation and is the idea behind minimum-weight perfect matching correction algorithms. More precisely, MWPM matches both kinds of anyons independently of each other and thus ignores potential correlations between σ^x - and σ^z -errors.

Decoherence models in which each qubit independently is subject to the channel

$$\rho \mapsto p_I \rho + p_x \sigma^x \rho \sigma^x + p_y \sigma^y \rho \sigma^y + p_z \sigma^z \rho \sigma^z \quad (1)$$

(with $p_I + p_x + p_y + p_z = 1$) allow for efficient simulation on a classical computer. While physical decoherence models may not exactly have the form of Eq. (1), they may be approximated by such a channel through a *Pauli twirl approximation* [16, 17]. The two most frequently studied noise models of the form of Eq. (1) are *independent bit- and phase-flip errors* ($p_x = p_b(1 - p_p)$,

$p_z = p_p(1 - p_b)$ and $p_y = p_b p_p$ for independent bit- and phase-flip probabilities p_b and p_p) and *depolarizing noise* ($p_x = p_z = p_y = \frac{p}{3}$). The theoretical maximal error rates up to which error correction is possible by exact error correction are known to be $p_b, p_p < 10.9\%$ for independent bit- and phase-flip errors [5] and $p < 18.9\%$ for depolarizing noise [18]. Any approximate error correction algorithm will yield threshold error rates below these theoretical maxima. For error rates below the threshold values, the probability of a logical error decreases exponentially with L .

Our algorithm is based on an analytically exact rewriting of the probability of each equivalence class which allows evaluation with the Metropolis algorithm. Let us discuss depolarizing noise here and note that our discussion generalizes straightforwardly to arbitrary error models of the form of Eq. (1). We have defined a depolarization rate p to mean that each spin has suffered a σ^x , σ^y , or σ^z error with probability $p/3$ each and no error with probability $1 - p$. Consequently, the probability of an error chain involving n single-qubit errors is up to a normalization constant given by $\left(\frac{p/3}{1-p}\right)^n \equiv e^{-\bar{\beta}n}$, where $\bar{\beta}$ is defined through

$$\bar{\beta} = -\log\left(\frac{p/3}{1-p}\right). \quad (2)$$

Given an anyon configuration A , the relative probability of equivalence class E can be written as

$$Z_E(\bar{\beta}) = \sum_E e^{-\bar{\beta}n}, \quad (3)$$

where the sum runs over all error chains that are compatible with the anyon configuration A and elements of equivalence class E , and n denotes the number of single-qubit errors in a particular error chain. The goal is to find the equivalence class E with maximal $Z_E(\bar{\beta})$.

The Metropolis algorithm allows us to approximate expressions of the form

$$\langle f(n) \rangle_{\beta, E} := \frac{\sum_E f(n) e^{-\beta n}}{Z_E(\beta)} \quad (4)$$

(we use β to denote a generic “inverse temperature” and $\bar{\beta}$ to denote the specific one defined through Eq. (2)). The sum is here over all error configurations in equivalence class E that are compatible with the syndrome information A . In order to approximate an expression of the form in Eq. (4), we pick one stabilizer at random and calculate the number Δn by which the total number of errors n in the code would change if that stabilizer were applied. If $\Delta n \leq 0$, we apply the stabilizer and if $\Delta n > 0$ we apply it with probability $e^{-\beta \Delta n}$. Summing up $f(n)$ over all steps and dividing by the total number of steps then yields our approximation to Eq. (4).

Deforming error patterns only through the application of stabilizers ensures that all error patterns in one

Markov chain belong to the same class, and that all of them are compatible with the same anyon configuration A . Since we will need the average $\langle f(n) \rangle_{\beta, E}$ for each equivalence class E , we need an initial error configuration from each equivalence class which is compatible with the measured anyon syndrome A . In fact, we will start the Metropolis Markov chains with the minimum weight error configuration from each equivalence class. The reason for starting with the minimum weight error configuration rather than a random initial configuration from the same equivalence class is based on the intuition that “heating up” from the groundstate to inverse temperatures β as needed for the equilibrium averages in Eq. (4) takes less time than “cooling down” from a high energy configuration. In order to find the minimum weight hypotheses from each equivalence class, first note that each hypothesis about what errors have happened has to link each p-anyon with a chain of σ^x errors to another p-anyon or to the corresponding type of boundary (left or right in Fig. 1), and similarly with σ^z errors for s-anyons (which can be linked to the top or bottom boundary). A qubit which has suffered a σ^y error is then simply one on which a σ^x and a σ^z error have occurred.

For a graph with weighted edges between an even number of vertices, Edmond’s MWPM algorithm [20] finds the pairing of minimal weight efficiently. We employ the library **Blossom V** [21] to perform MWPM. For the graphs which are relevant for the surface code, MWPM can be performed in runtime complexity $O(L^2)$ and can be parallelized to $O(L^0)$ [13]. In order to obtain such graphs, we assign to the edge between two anyons (corresponding to the vertices) the minimal number of single-qubit errors necessary to link them (their Manhattan distance). For each anyon, we place a virtual partner on the closer boundary of the type which is able to absorb it (top and bottom for s-anyons and left and right for p-anyons). We then add an edge between each anyon and its virtual partner (with weight again given by the Manhattan distance) and zero-weight edges between all virtual anyons on the same boundary. Including these virtual anyons ensures that the number of vertices in the graph is even and that each anyon can be matched to the closest absorbing boundary. The zero-weight edges ensure that unnecessary virtual anyons can be removed at no cost. The four equivalence classes of errors in the surface code may be identified by determining the parity of the number of errors that lie on a given line that links the top and bottom or left and right boundary. In order to find the minimum weight error configuration of all four equivalence classes, we need be able to enforce changes in these parities. This can be achieved by adding one further virtual anyon on the top and bottom or the left and right boundary, respectively, and connecting it with zero-weight edges to all virtual anyons already present at this boundary [19]. It is thus guaranteed that a any pairing of all real and virtual anyons of the same type is

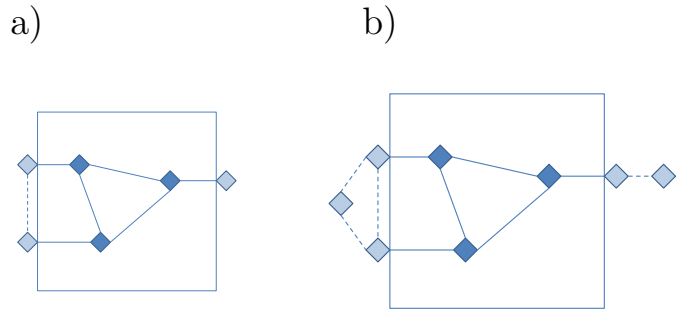


FIG. 2. Three p-anyons (solid squares) have been detected in a surface code. In order to find hypotheses of minimum weight (maximal probability) about what errors have occurred, we first add virtual anyons (light squares) on the closest absorbing boundary of each real anyon and connect virtual anyons residing on the same boundary; see the a) part of the figure. Dashed lines represent zero-weight edges, solid lines represent non-zero-weight edges. In part b) of the figure, we place an additional virtual anyon on the left and right boundary. Note that each possible pairing in part b) is an element of a different equivalence class than the pairings which are possible in part a).

an element of a different equivalence class than the one obtained if these two additional virtual anyons had not been included. See Fig. 2 for an illustration.

Note that \sum_E in Eq. (4) has $2^{n_{\text{stab}}}$ summands for each equivalence class E , so knowing an averaged sum is as good as knowing the whole sum. We have

$$Z_E(\bar{\beta}) = \left\langle e^{-\bar{\beta}n} \right\rangle_{\beta=0, E} \times 2^{n_{\text{stab}}} , \quad (5)$$

corresponding to a simple Monte Carlo sampling of the sum. However, the sum is dominated by an exponentially small fraction of summands with “energy” n close to the minimal value, so Monte Carlo sampling is computationally similarly expensive as a brute force calculation of the sum. Our goal is thus to rewrite $Z_E(\bar{\beta})$ in a way that involves only quantities which are evaluable efficiently with the Metropolis algorithm. Applying the fundamental theorem of calculus we have

$$\begin{aligned} \log Z_E(\bar{\beta}) &= \int_0^{\bar{\beta}} d\beta \partial_\beta \log Z_E(\beta) + \log Z_E(\beta=0) \\ &= - \int_0^{\bar{\beta}} d\beta \langle n \rangle_{\beta, E} + n_{\text{stab}} \log 2 . \end{aligned} \quad (6)$$

If we know the functions $\langle n \rangle_{\beta, E}$, the most likely equivalence class is, according to Eq. (6), the one in which the area under the curve is smallest. In the correctable regime ($p < p_c$) the differences in “free energy”

$$\begin{aligned} F_E(\bar{\beta}) &= -\frac{1}{\bar{\beta}} \log Z_E(\bar{\beta}) \\ &= \frac{1}{\bar{\beta}} \int_0^{\bar{\beta}} d\beta \langle n \rangle_{\beta, E} + \text{const} \end{aligned} \quad (7)$$

between the different equivalence classes grow proportionally in L and correspondingly the probability of all equivalence classes but the most likely one decreases exponentially with L . However, as the values of the integrand $\langle n \rangle_{\beta,E}$ grow with L^2 , the relative differences in free energy *decrease* with L .

For a positive β , the average number of errors $\langle n \rangle_{\beta,E}$ can be efficiently calculated to arbitrary accuracy by means of the Metropolis algorithm. The integral $\int_0^{\bar{\beta}} d\beta$ can be calculated to arbitrary accuracy by first calculating the values $\langle n \rangle_{\beta,E}$ for a sufficient number of inverse temperatures β and then applying a quadrature formula like Simpson's Rule. We conclude that the free energy $F_E(\bar{\beta})$ and hence the probability of each equivalence class can be efficiently calculated to arbitrary accuracy – in contrast to the idea that error correction algorithms are either inefficient or suboptimal.

THE SINGLE TEMPERATURE ALGORITHM

Recall that we are not interested in the precise value of the integrals $\int_0^{\bar{\beta}} d\beta \langle n \rangle_{\beta,E}$, but only in knowing for which equivalence class E this integral is smallest. For this reason, calculating the whole integral is quite often an overkill. In fact, most of the relevant information contained in the function $\langle n \rangle_{\beta,E}$ can be extracted by finding its value for a *single* inverse temperature β^* .

Assume that we determine the values $\langle n \rangle_{\beta^*,E}$ for some $\beta^* > 0$ for all equivalence classes E . If the functions $\langle n \rangle_{\beta,E}$ for the different equivalence classes do not cross, knowing the values $\langle n \rangle_{\beta^*,E}$ is as good as knowing the whole integrals $\int_0^{\bar{\beta}} d\beta \langle n \rangle_{\beta,E}$ for deciding for which equivalence class E the integral is smallest.

As $\beta \rightarrow 0$, each qubit is affected by an x -, y -, or z -error or no error at all with probability $\frac{1}{4}$, so $\langle n \rangle_{\beta,E} \rightarrow \frac{3}{4}n_{\text{qubits}}$, where $n_{\text{qubits}} = n_{\text{stab}} + 1$ is the number of data qubits in the code. The low- β tail of the function $\langle n \rangle_{\beta,E}$ thus contains almost no information about the equivalence class E . So while the integral $\int_0^{\bar{\beta}} d\beta \langle n \rangle_{\beta,E}$ is dominated by its low- β part ($\langle n \rangle_{\beta,E}$ is a monotonically decreasing function of β), the *differences* between these integrals for the different equivalence classes are mainly due to their high- β part. So even if there are crossings in the low- β tails of the functions $\langle n \rangle_{\beta,E}$, basing the decision for the most likely equivalence class on a single value $\langle n \rangle_{\beta^*,E}$ is likely to yield the same outcome as basing the decision on the whole integral $\int_0^{\bar{\beta}} d\beta \langle n \rangle_{\beta,E}$. We thus define our *single-temperature algorithm* as sampling the values $\langle n \rangle_{\beta^*,E}$ for all equivalence classes and performing error correction in accordance with the equivalence class E for which this value is smallest.

This algorithm has only two free parameters, namely β^* and the number of steps for which we perform the

Metropolis algorithm in order to sample $\langle n \rangle_{\beta^*,E}$. Note that for $\beta^* \rightarrow 0$ the single-temperature algorithm becomes useless ($\langle n \rangle_{\beta^*=0,E} = \frac{3}{4}n_{\text{qubits}}$ for all equivalence classes) and for $\beta^* \rightarrow \infty$ the single-temperature algorithm coincides with MWPM as it bases its decision on the minimum-weight error chain of each equivalence class. So if the single-temperature algorithm is able to outperform MWPM, it has to be based on $\beta^* \approx \bar{\beta}$. Empirically, we find that in the case of depolarizing noise choosing a value for β^* different from $\bar{\beta}$ does not offer much room for improvement, so we set $\beta^* = \bar{\beta}$ for this error model. As for the second parameter, we may ask how many Metropolis steps are necessary for our single-temperature algorithm to achieve logical error rates below those achievable with MWPM.

In order to set the bar high, we compare our algorithm with a somewhat enhanced version of MWPM. This uses our above described method of enforcing changes in the equivalence class for the MWPM algorithm, thus allowing us to find two non-equivalent minimum-weight error chains for both kinds of anyons. Combining these 2×2 error chains gives four hypotheses about the errors that have happened. Each is the most likely within its equivalence class for an approximate error model where the correlations between x - and z -errors are ignored (the only one MWPM can treat straightforwardly). We can then determine which of the four is most likely overall according to the true depolarizing noise model – i.e., determine the hypotheses with the minimal number of errors, where one x - and one z -error on the same qubit count as one y -error. As we show below, this enhanced version of MWPM indeed leads to lower logical error rates as standard MWPM (as performed for example in Ref. [22]), as it does not completely ignore correlations between x - and z -errors [23].

Fig. 3 shows the number of Metropolis steps which are necessary for our single-temperature algorithm to achieve the logical error rates achievable with enhanced MWPM, and illustrates that already an $O(L^3)$ runtime is sufficient for this. The inset in Fig. 3 shows the logical error rates achievable with our single-temperature algorithm if the quantities $\langle n \rangle_{\beta^*,E}$ are sampled over $10L^3$ Metropolis steps. For values of p below 15%, the logical error rate decreases exponentially with L , meaning that the threshold error rate for our algorithm is significantly below the theoretical maximum of 18.9% [18] and the value of 18.5% achieved in Ref. [14] but closer to threshold of MWPM [22]. However, the relevant figures of merit in practice are the logical error rates achievable well below threshold (where our algorithm offers significant improvement over MWPM, see below) and the runtime complexity (where our algorithm offers significant improvement over the algorithm of Ref. [14]).

The quantity which determines whether error correction will be successful is the difference $\min \langle n \rangle_{\beta^*,\text{false}} - \langle n \rangle_{\beta^*,\text{true}}$, where $\min \langle n \rangle_{\beta^*,\text{false}}$ denotes the minimal av-

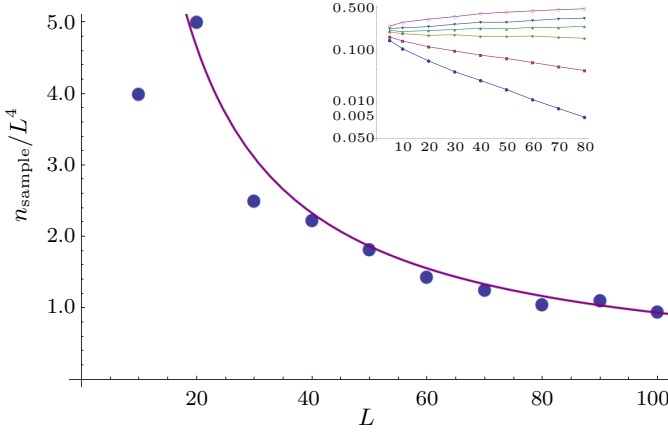


FIG. 3. The plot shows n_{sample}/L^4 against L for the case of depolarizing noise with $p = 14\%$. Here n_{sample} is the minimum number of Metropolis steps over which the quantities $\langle n \rangle_{\beta^*, E}$ (with $\beta^* = \bar{\beta}$) must be sampled for each equivalence class E , in order for the logical error rate of the single temperature algorithm to be below that of enhanced MWPM. Each data point (shown in blue) is averaged over as many error configurations as were required for 2000 logical errors to occur. The ratio n_{sample}/L^4 clearly does not diverge. Indeed, the agreement with the function $0.93L^{-1.00}$ (purple line) shows that it seems set to vanish as $L \rightarrow \infty$. This shows that the complexity of $n_{\text{sample}} \in O(L^4)$, and appears to be as low as $n_{\text{sample}} \in O(L^3)$. The inset shows the logical error rates of our single-temperature algorithm alone with $n_{\text{sample}} = 10L^3$, for various system sizes L (horizontal axis) and depolarization rates p (from bottom to top $p = 13\%, 14\%, 15\%, 15.5\%, 16\%, 17\%$).

eraged number of errors of all three false equivalence classes. This difference is displayed for various values of p and L in Fig. 4. For $p \lesssim 16\%$ this difference increases linearly with L , while for $p = 17\%$ it becomes even negative for large enough L . This is to be expected: for an error rate sufficiently close to the 18.9% threshold, each of the averages $\langle n \rangle_{\beta^*, E}$ for the four equivalence classes E has the same probability for being the smallest one, such that the probability that one of the three false equivalence classes becomes minimal approaches $\frac{3}{4}$.

Fig. 5 compares the logical error rates achievable with different algorithms for $L = 15$ and different depolarization rates p . We set the logical error rates achievable with standard MWPM to unity and divide them by the logical error rates achievable with alternate algorithms. The algorithms which are displayed are:

- A) Standard, unimproved MPWM, as employed in Ref. [22]. An x - and a z -error on the same qubit count as two errors.
- B) Enhanced MWPM. An x - and a z -error on the same qubit count as two errors during the matching, but as one error during a final comparison of all equivalence classes.
- C) Single-temperature algorithm with quantities

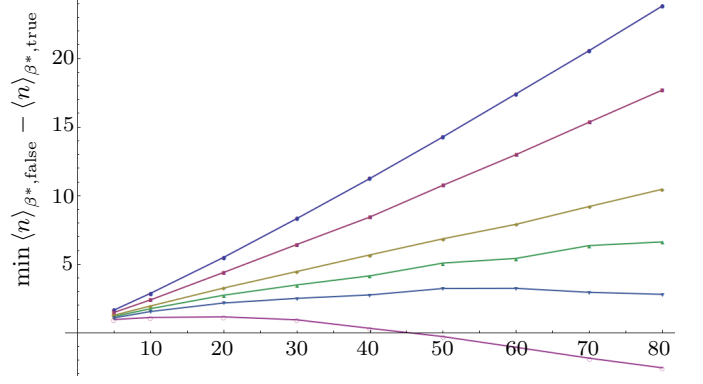


FIG. 4. The plot shows the distinguishabilities $\min \langle n \rangle_{\beta^*, \text{false}} - \langle n \rangle_{\beta^*, \text{true}}$ (vertical axis; we use $\beta^* = \bar{\beta}$) for various depolarization rates p and code sizes L (horizontal axis). Different lines correspond to different depolarization rates p . From top to bottom we have $p = 13\%, 14\%, 15\%, 15.5\%, 16\%, 17\%$. All quantities $\langle n \rangle_{\beta^*, E}$ are sampled over $10L^3$ Metropolis steps and each data point is averaged over as many error configurations as are necessary to obtain 2000 logical errors.

$\langle n \rangle_{\beta^*, E}$ sampled over $10L^3$ Metropolis steps.

- D) Calculation of the entire integral $\int_0^{\bar{\beta}} d\beta \langle n \rangle_{\beta, E}$ through first sampling $\langle n \rangle_{\beta, E}$ over $10L^3$ Metropolis steps for 21 equidistant values of β ($0, \frac{\bar{\beta}}{20}, \dots, \bar{\beta}$) and then applying Simpson's quadrature formula.
- E) Single-temperature algorithm with quantities $\langle n \rangle_{\beta^*, E}$ sampled over $100L^3$ Metropolis steps.
- F) The parallel-tempering algorithm developed in Ref. [14].

The logical error rates decrease from A to F, while the runtime complexities increase (with the exception of algorithm E requiring less time than algorithm D). We see that the advantages achievable over algorithm A vanish as $p \rightarrow p_c = 18.9\%$ but increase the lower p gets. Note that algorithm D shows only a marginal advantage over algorithm C, as expected. We believe our single-temperature algorithm (C and E) to offer the most attractive trade-off ratio between low logical error rates and low classical runtime complexity, as they increase the latter only modestly compared with algorithms A and B, while algorithm F has a super-polynomial (in L) runtime complexity. Algorithm D is strictly dominated by algorithm E (lower runtime and lower logical error rate).

Fig. 6 shows the logical error rates of algorithm A divided by those of algorithm C for various values of p and L . We see the advantage of algorithm C increasing for lower p and larger L . Note that we expect real quantum computers to be operated at error rates p below and code distances L above those displayed in Fig. 6.

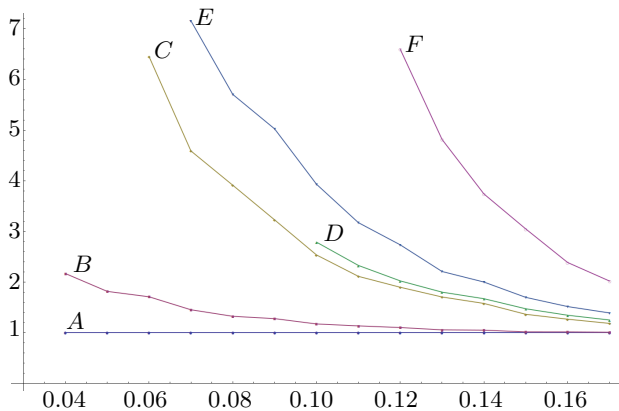


FIG. 5. The plots show the logical error rate of MWPM (algorithm A in the main text) divided by the logical error rate of algorithms A to F described in the main text, for various depolarization rates p (horizontal axis) and a code of linear size $L = 15$. Each data point is averaged over as many error configurations as are necessary to obtain 2000 logical errors. A value greater than 1 denotes an increase in effectiveness over MWPM, with greater increases for higher values.

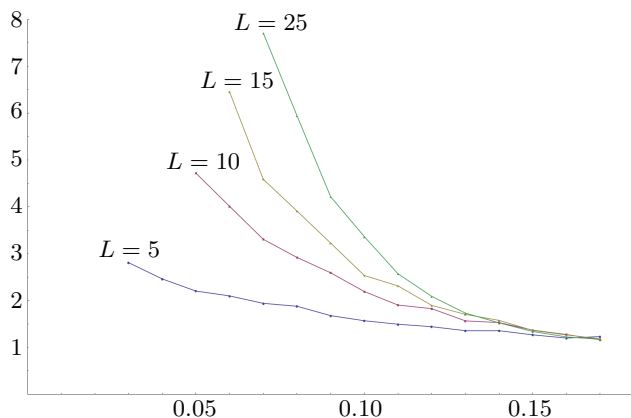


FIG. 6. The plots show the logical error rate of algorithm A divided by the logical error rate of algorithm C for various depolarization rates p (horizontal axis) and code sizes L . Each data point is averaged over as many error configurations as are necessary to obtain 2000 logical errors.

To get an idea about what effect this reduction of the logical error rates has on the necessary code size, let us have a look at the code size which is needed for a proof of principle experiment. I.e., given some physical error rate p , which code size L is needed to bring the logical error rate below the physical one? For $p = 13\%$, we need $L \geq 6$ in order to achieve a logical error rate below p with algorithm C and need $L \geq 15$ with algorithm A. So by modestly enhancing the classical runtime complexity, we are able to reduce the number of physical data qubits required for such a proof of principle experiment from 421 to 61.

The advantage of algorithm C over algorithms A and

B is due to at least two different reasons. First, MWPM is naturally suited to the error model of independent bit- and phase-flips, where each type of error can be treated independent of the other. It is much less suited to error models such as depolarizing noise which feature correlations between bit- and phase-flip errors. Algorithm B can only partially overcome these limitations. Second, while algorithms A and B are based on finding the most likely error chain and hoping that it is an element of the most likely equivalence class, algorithms C to F are based on finding the most likely equivalence class. An interesting question is thus how our single-temperature algorithm compares with MWPM for independent bit- and phase-flip errors, the error model MWPM is best suited to and where only the second advantage applies. Empirically, we find that for this error model choosing $\beta^* = 0.85\beta$ works best and that for a bit-/phase-flip probability of 10% (close to the theoretical threshold) $O(L^4)$ Metropolis steps are again sufficient to achieve a logical error rate below the one of MWPM, with $0.38L^{3.77}$ giving the best fit to the required number. For $L = 30$, a bit-/phase-flip probability of 8%, and by sampling over $10L^4$ Metropolis steps we achieve a logical error rate which is a factor 1.3 lower than the one of MWPM. So significant improvements over MWPM can be achieved even for the error model best suited to it, though the advantage is much more drastic for an error model with correlations between bit- and phase-flip errors, with which our single-temperature algorithm can deal very naturally.

Also note that there are quantum error correction codes such as the toric code with higher-than-two-dimensional qudits to which MWPM is not applicable. In contrast, the algorithm presented here is readily generalizable to any kind of stabilizer code.

PARALLELIZATION

The runtime of our algorithm can be reduced by a factor $O(L^2)$ using parallelization which exploits the fact that our algorithm needs local changes only. We partition the whole code into $O(L^2)$ rectangles of area $O(L^0)$. Adjacent rectangles overlap along lines of data qubits, while qubits in the corners belong to four rectangles, see Fig. 7. The rectangles are collected into four groups 0, 1, 2, 3 such that the rectangles within one group have no overlapping qubits. At step i of the Metropolis Markov chain, we choose one stabilizer in each rectangle belonging to group $(i \bmod 4)$ at random and probe whether to apply it or not according to the Metropolis procedure. This way we can guarantee that no data qubit is affected by more than one applied stabilizer at each step. If a randomly chosen stabilizer is applied and flips a qubit which is shared with an other rectangle (other rectangles), this flip is communicated to the adjacent rectangle(-s). For each rectangle, we add up the number of local errors n

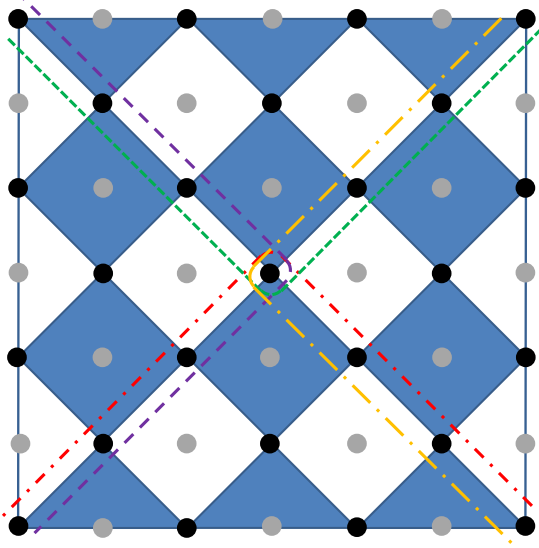


FIG. 7. The dashed lines partition the code into rectangles of size $O(1)$. Data qubits along the boundaries belong to two rectangles and data qubits in the corners to four.

over the different Metropolis steps and calculate the total average $\langle n \rangle_{\beta^*, E}$ in the end. To compensate for double-(quadruple-)counting, errors on qubits along the boundary thereby have to be discounted by a factor $\frac{1}{2}$ and errors on qubits in the corners by a factor $\frac{1}{4}$. As we probe now $O(L^2)$ stabilizers in each time step, the runtime reduces from $O(L^\alpha)$ to $O(L^{\alpha-2})$.

IMPERFECT STABILIZER MEASUREMENTS

Let us assume that each stabilizer measurement yields the wrong result with probability p_M . If stabilizers are measured by use of CNOT gates, this model is a simplification in that it ignores correlations between physical qubit errors (induced by the channel Eq. (1)) and measurement errors [24]. In order to make the failure probability small despite non-negligible p_M , we now necessarily need to perform stabilizer measurements at several times $t = 1, 2, \dots, t_{\max}$. A hypothesis about what errors have happened then not only has to state which data qubit has suffered an error in which time interval $[t, t+1]$, but also which stabilizer measurement has been erroneous at which time t . Such hypotheses can be deformed into equivalent ones by applying a bit-/phase-flip σ^x/σ^z to a particular qubit at time intervals $[t-1, t]$ and $[t, t+1]$ and inverting the hypothesis about whether the stabilizer measurements at time t that anti-commute with this error have been erroneous (see the illustration in Fig. 8).

In the case of depolarizing noise, a hypothesis that involves n data-qubit errors and m erroneous syndrome

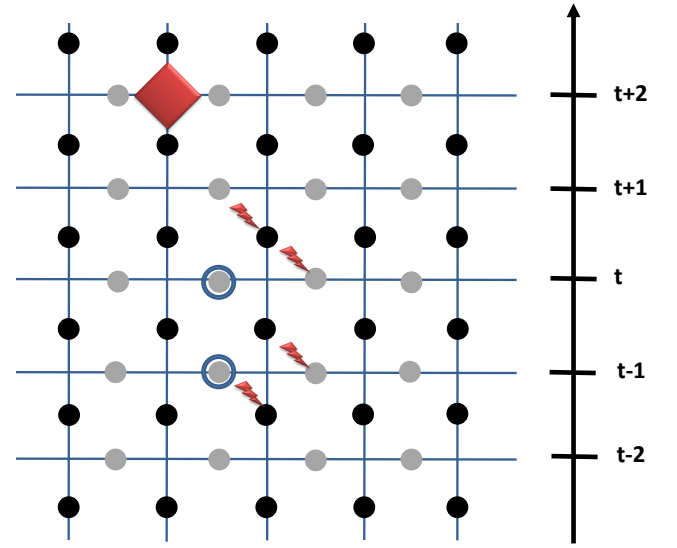


FIG. 8. Only one type of errors and stabilizers and a single chain of data qubits (black dots) and syndrome qubits (grey dots) are depicted here for simplicity. Time runs from bottom to top. Stabilizer operators are measured at times $\dots, t-1, t, t+1, \dots$. The two encircled syndrome qubits have detected errors (-1 eigenvalues). A possible hypothesis is that no data qubit has suffered an error and both syndrome measurements have been erroneous. An alternate hypothesis would state that the data qubits indicated by red lightning bolts have suffered errors and that syndrome measurement by use of the syndrome qubits indicated by red lightning bolts has been erroneous. Different equivalent hypotheses can be deformed into each other through the application of operators like the red square that invert whether a hypothetical error has happened or not at two adjacent syndrome qubits and one data qubit at two subsequent times.

measurements has a relative probability

$$\left(\frac{p/3}{1-p} \right)^n \times \left(\frac{p_M}{1-p_M} \right)^m \equiv \exp[-\bar{\beta}(n + \xi m)] \quad , \quad (8)$$

where $\bar{\beta}$ is as defined in Eq. (2) and

$$\xi = \frac{1}{\bar{\beta}} \log \frac{1-p_M}{p_M} \quad . \quad (9)$$

The “energy” of a hypothesis is thus given by $n + \xi m$ where ξ determines the relative weight of erroneous syndrome measurements to data qubit errors. Our method to find the most probable equivalence class in the case of perfect stabilizer measurements can thus be generalized to the case where syndrome measurements fail with a considerable probability. Numerical results for the latter case will appear in future work.

CONCLUSIONS

We have shown that adding runtime complexity $O(L^\varepsilon)$, with $\varepsilon < 2$, to the fastest surface code error correction

algorithms [12, 13] allows us to attain logical error rates significantly below those achievable with MWPM. More precisely, it appears that $\varepsilon \in [1.00, 1.77]$ is sufficient for error models which are mixtures of depolarizing noise and independent bit- and phase-flip errors. This means that in order to reach a certain target logical error rate, the surface code may be significantly smaller if our algorithm is applied. While for error rates close to the threshold, where error correction is almost infeasible, both algorithms perform similarly, the advantage of our single-temperature algorithm over MWPM increases as error rates are decreased (and code sizes are increased). This makes it particularly relevant in practice, as quantum computers will be operated at error rates significantly below threshold, and the ability to use smaller code sizes will have a significant practical advantage.

ACKNOWLEDGMENTS

This work is supported by the Swiss NSF, NCCR Nanoscience, NCCR QSIT, and IARPA.

-
- [1] J. R. Wootton, *Journal of Modern Optics* **20**, 1717 (2012), arXiv:1210.3207.
 - [2] A. Yu. Kitaev, *Annals Phys.* **303**, 2 (2003), arXiv:quant-ph/9707021.
 - [3] M. H. Freedman, A. Y. Kitaev, M. J. Larsen, and Z. Wang, (2002), quant-ph/0101025.
 - [4] S. B. Bravyi and A. Y. Kitaev, quant-ph/9811052 (1998).
 - [5] E. Dennis, A. Y. Kitaev, A. Landahl, and J. Preskill, *J. Math. Phys.* **43**, 4452 (2002), quant-ph/0110143.
 - [6] R. Raussendorf and J. Harrington, *Phys. Rev. Lett.* **98**, 190504 (2007), quant-ph/0610082.
 - [7] A. G. Fowler, A. M. Stephens, and P. Groszkowski, *Phys. Rev. A* **80**, 052312 (2009), arXiv:0803.0272.
 - [8] F. L. Pedrocchi, S. Chesi, and D. Loss, *Phys. Rev. B* **83**, 115415 (2011), arXiv:1011.3762.
 - [9] D. P. DiVincenzo and F. Solgun, quant-ph/1205.1910 (2012).
 - [10] S. E. Nigg and S. M. Girvin, arXiv:1212.4000 (2012).
 - [11] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, *Phys. Rev. A* **86**, 032324 (2012), arXiv:1208.0928.
 - [12] G. Duclos-Cianci and D. Poulin, *Phys. Rev. Lett.* **104**, 050504 (2010), arXiv:0911.0581.
 - [13] A. G. Fowler, A. C. Whiteside, and L. C. L. Hollenberg, *Phys. Rev. Lett.* **108**, 180501 (2012), arXiv:1110.5133.
 - [14] J. R. Wootton and D. Loss, *Phys. Rev. Lett.* **109**, 160503 (2012), arXiv:1202.4316 (2012).
 - [15] As the error correction algorithm of Ref. [12] itself allows for trade-offs between runtime and error rates, and MWPM based algorithms have attained most interest in the literature, we do not directly compare our algorithm to the former.
 - [16] W. Dür, M. Hein, J. I. Cirac, and H.-J. Briegel, *Phys. Rev. A*, **72**, 052326 (2005).
 - [17] J. Emerson, M. Silva, O. Moussa, C. Ryan, M. Laforest, J. Baugh, D. G. Cory, and R. Laflamme, *Science*, **317**, 1893 (2007).
 - [18] H. Bombin, R. S. Andrist, M. Ohzeki, H. G. Katzgraber, and M. A. Martin-Delgado, *Phys. Rev. X* **2**, 021004 (2012), arXiv:1202.1852.
 - [19] If there are no virtual anyons on a certain boundary, we connect the additional virtual anyon to all *real* anyons of the same type, with edges weighted by their distance to the corresponding boundary. If there are no real anyons, we connect the two new virtual anyons placed on opposite boundary. This way, the minimum weight error configuration of each equivalence class can be found for any possible anyon configuration.
 - [20] J. Edmonds, *Canad. J. Math.* **17**, 449 (1965).
 - [21] V. Kolmogorov, *Math. Prog. Comp.*, 1:43, 2009.
 - [22] D. S. Wang, A. G. Fowler, A. M. Stephens, and L. C. L. Hollenberg, *Quantum Inf. and Comput.* **10**, 456 (2010), arXiv:0905.0531.
 - [23] Quite surprisingly, we have found regimes (e.g., $L = 25$ and $12\% \leq p < p_c$) where “enhanced” MWPM seems to perform slightly worse than standard MWPM.
 - [24] D. S. Wang, A. G. Fowler, and L. C. L. Hollenberg, *Phys. Rev. A* **83**, 020302(R) (2011), arXiv:1009.3686.